The 2020 Bilingual, Bi-Directional WebNLG+ Shared Task Overview and Evaluation Results (WebNLG+ 2020)

Thiago Castro Ferreira, Claire Gardent, Nikolai Ilinykh, Chris van Der Lee, Simon Mille, Diego Moussallem, Anastasia Shimorina

The 2020 Bilingual, Bi-Directional WebNLG+ Shared Task Overview and Evaluation Results (WebNLG+ 2020)

Thiago Castro Ferreira[1], **Claire Gardent**[2], **Nikolai Ilinykh**[3], **Chris van der Lee**[4], **Simon Mille**[5], **Diego Moussallem**[6], **Anastasia Shimorina**[7]
[1] Federal University of Minas Gerais, Brazil [2] CNRS / LORIA, France
[3] University of Gothenburg, Sweden [4] Tilburg University, The Netherlands
[5] Universitat Pompeu Fabra, Spain [6] Paderborn University, Germany
[7] Université de Lorraine / LORIA, France
webnlg-challenge@inria.fr

Abstract

WebNLG+ offers two challenges: *(i)* mapping sets of RDF triples to English or Russian text (generation) and *(ii)* converting English or Russian text to sets of RDF triples (semantic parsing). Compared to the eponymous WebNLG challenge, WebNLG+ provides an extended dataset that enable the training, evaluation, and comparison of microplanners and semantic parsers. In this paper, we present the results of the generation and semantic parsing task for both English and Russian and provide a brief description of the participating systems.

1 Introduction

The motivation behind the WebNLG challenges is twofold. On the one hand, we seek to provide a common benchmark on which to evaluate and compare "micro-planners", i.e., Natural Language Generation (NLG) systems which can handle the full range of micro-planning tasks including document structuring, aggregation, regular expression generation, lexicalisation and surface realisation (Reiter and Dale, 2000). On the other hand, we are interested in building connections with research from the semantic web community which explores the relationship between knowledge bases (KBs) and natural language. There is a clear parallel between open information extraction (Open IE) and RDF-based semantic parsing, and between RDF-to-Text generation and KB verbalisation. Yet the interaction between NLP and Semantic Web research remains limited. By highlighting the NLP tasks involved in mapping RDF triples and natural language, we aim to stimulate cross-fertilisation between NLP and Semantic Web research.

WebNLG datasets align sets of RDF triples with text. While the 2017 WebNLG shared task required participating systems to generate English text from a set of DBpedia triples (Gardent et al., 2017b), the 2020 WebNLG+ challenge additionally includes generation into Russian and semantic parsing of English and Russian texts. Thus the WebNLG+ challenge encompasses four tasks: RDF-to-English, RDF-to-Russian, English-to-RDF and Russian-to-RDF.

Timeline. The training and development data was released on April 15, 2020, preliminary evaluation scripts on April, 30th and final evaluation scripts on May, 30th. The test data was made available on September, 13th and the deadline for submitting system results was September, 27th. Automatic evaluation results were announced on October, 9th and the first version of the human evaluation results on November, 20th. The final version of the human evaluation results were released on November, 26th. Results were first released anonymously so that participants had the opportunity to withdraw their systems.

In what follows, we summarise the main features of WebNLG+ 2020. Section 2 describes the datasets used for the challenge. Section 3 presents the participating systems. Section 4 introduces the evaluation methodology, Section 5 discusses the participants results in the automatic evaluation and Section 6 in the human evaluation. Finally, Section 7 depicts the correlations between automatic evaluation metrics and human ratings as well as Section 8 concludes with pointers for further developments.

2 Data

2.1 English WebNLG

The English challenge data uses the version 3.0[1] of the WebNLG corpus (Gardent et al., 2017a). This version has undergone some significant changes

[1]For versioning see here: https://gitlab.com/shimorina/webnlg-dataset

	Train	Dev	Test (D2T)	Test (SP)
RDF triple sets	13,211	1,667	1,779	752
Texts	35,426	4,464	5,150	2,155
Properties	372	290	220	201

Table 1: WebNLG 3.0 English data statistics. Properties: the number of unique DBpedia properties.

compared to the data used in WebNLG'2017. The training data in 2020 consists of 16 DBpedia categories:

- the 10 seen categories used in 2017: Airport, Astronaut, Building, City, ComicsCharacter, Food, Monument, SportsTeam, University, and WrittenWork;
- the 5 unseen categories of 2017 that became part of the seen data in 2020: Athlete, Artist, CelestialBody, MeanOfTransportation, Politician;
- one new category that was added to the training set (Company).

The following data improvements were also carried out: *(i)* around 5,600 texts were cleaned from misspellings, and missing triple verbalisations were added to some texts; *(ii)* information about tree shapes and shape types were added to each RDF tree; *(iii)* some properties were unified to ensure consistency across the corpus. Table 1 shows some dataset statistics. Training and developments sets were the same for the data-to-text (D2T) and semantic parsing (SP) tasks, unlike the test sets which are different for the two tracks.

New test sets were also collected for English because the previous test set has been made public. Following the tradition of several test data types, introduced in the previous shared task (Gardent et al., 2017b), we kept them in this year edition and introduced one new type *unseen entities*. The three types of the test data are:

- seen categories: RDF triples based on the entities and categories seen in the training data (e.g., *Alan Bean* in the category Astronaut);
- unseen entities: RDF triples based on the categories seen in the training data, but not entities (e.g., *Nie Haisheng* in the category Astronaut);
- unseen categories: RDF triples based on the categories not present in the training data. Three unseen categories were introduced in this year edition: Film, Scientist, and MusicalWork. Out of 220 unique properties in the test set for the D2T task, 39 properties were never seen in the training and development data.

	D2T (RDF)	SP (texts)
Seen categories	490 (28%)	606 (28%)
Unseen entities	393 (22%)	457 (21%)
Unseen categories	896 (50%)	1,092 (51%)
Total	1,779	2,155

Table 2: Number of RDF triple sets and texts by test set type for data-to-text and semantic parsing respectively.

Statistics of the test splits are shown in Table 2. Unlike the test set, the development set included data from seen categories only. However, participants were notified about the inclusion of unseen data from the beginning of the challenge and had to model the unseen data scenario by their own means.

New data for WebNLG-3.0 was collected with Amazon Mechanical Turk, and some triple verbalisations (part of the Film category) were done by students. For crowdsourcing, we followed the same procedure as was followed for the collection of the initial WebNLG data (Gardent et al., 2017a), but without the verification step. Instead, after collection, a spellchecker and quality checks were run and, if problems were spotted, texts were edited manually. Quality checks mainly consisted in verifying if triple entities are present in texts. We collected around three references per RDF triple sets.

2.2 Russian WebNLG

Russian WebNLG was translated from English WebNLG for nine DBpedia categories: Airport, Astronaut, Building, CelestialBody, ComicsCharacter, Food, Monument, SportsTeam, and University. Table 3 shows some statistics of the Russian dataset. For the test set, only the data of the *seen categories* type is present, which makes the Russian track much easier to handle for participating systems.

Russian data also possesses some additional features compared to the English data: links between English and Russian entities from subjects and verbal objects of RDF triples were given. Some of them were extracted from DBpedia between En-

	Train	Dev	Test (D2T)	Test (SP)
RDF triple sets	5,573	790	1,102	474
Texts	14,239	2,026	2,780	1,206
Properties	226	115	192	164

Table 3: WebNLG 3.0 Russian data statistics. Properties: the number of unique DBpedia properties.

glish and Russian entities by means of the property *sameAs* (e.g., (Spaniards, sameAs, испанцы)). For the entities without such a property, links were created manually. The links served as pointers for translators. During the test phase, those features were available for the RDF-to-text track.

The Russian data creation followed the procedure below:

1. Russian WebNLG was translated from the English WebNLG version 2.0 with the MT system of Sennrich et al. (2017), as described in Shimorina et al. (2019).

2. It was then post-edited using crowdsourcing on the Yandex.Toloka platform in two steps:

- we asked people to post-edit Russian texts given original English texts and provided them with some pointers for translation of entities (the links described above). Crowdworkers were asked to use the pointers as much as possible.
- given the post-edited sentences, we asked people to check if the text was translated properly (in terms of grammar, spelling, etc.) and if the entity translation was correct. If the translation was detected as erroneous, it was moved to the post-edit step again.

3. Afterwards, some sanity checks and a spellchecker were run to ensure data quality. All the detected cases were then manually verified by experts (Russian native speakers), and they edited the texts one more time if needed.

Based on this procedure, we assume that the Russian data is of a decent quality. However, based on manual inspections, some texts may still be lacking in terms of fluency and correctness. Note also that the Russian version was derived from the English WebNLG version 2.0, where some errors in semantic content realisation were present.

3 Participating Systems

The WebNLG+ data was downloaded more than 100 times, 17 teams submitted 48 system runs. From this sample, two teams withdrew their results, which gave us 15 participating teams with 46 runs for automatic evaluation (Table 4). For human evaluation, we evaluated 14 teams for English and 6 teams for Russian. Only one team participated in all four tasks (bt5). Two participants (Amazon AI (Shanghai) and CycleGT) submitted models for both generation and semantic parsing but only for English. All other submissions focused on generation, one only for Russian (med), five for English only (TGen, UPC-POE, RALI-Université de Montréal, ORANGE-NLG, NILC) and four for both Russian and English (cuni-ufal, FBConvAI, Huawei Noah's Ark Lab, OSU Neural NLG).

In what follows, we summarise the primary submissions of the 15 participating teams.

3.1 Monolingual, Mono-Task, Template-based Approaches

Among all system submissions, two of them used templates: RALI-Université de Montréal and DANGNT-SGU.

RALI-Université de Montréal. Lapalme (2020) implements a symbolic approach which captures the various substeps of NLG programmatically. The input set of RDF triples is partitioned and ordered into sentence sized subsets. Each subset is then transformed into a sentence using Python procedures designed to encode 200 manually defined sentence templates. Aggregation is handled by combining templates and referring expression generation by using names for first occurrences and pronouns for subsequent occurrences (within a template). The REAL surface realiser is used to map the resulting sequence of sentence templates to sentences.

DANGNT-SGU. Tran and Nguyen (2020) derive delexicalised templates from the data by replacing RDF subjects and objects with placeholders and identifying their text counterparts using the Jaro-Winkler similarlity metrics.

3.2 Mono-lingual, Mono-task, Neural Approaches

med. Blinov (2020) focuses on generation into Russian. They used the pre-trained Russian GPT-2 language model (Radford et al., 2019) augmented with a classification head and fine-tuned on the WebNLG+ RDF-to-Russian dataset. The author experimented with various sampling methods and

Team	Affiliation	Country	D2T		SP	
			En	Ru	En	Ru
med	Sber AI Lab	Russia	-	✓	-	-
RALI-Université de Montréal	Université de Montréal	Canada	✓	-	-	-
ORANGE-NLG	Orange Labs	France	✓	-	-	-
cuni-ufal	Charles University	Czechia	✓	✓	-	-
TGen	AIST	Japan	✓	-	-	-
bt5	Google	US	✓	✓	✓	✓
UPC-POE	Universitat Politècnica de Catalunya	Spain	✓	-	-	-
DANGNT-SGU	Saigon University	Vietnam	✓	-	-	-
Huawei Noah's Ark Lab	Huawei Noah's Ark Lab	UK	✓	✓	-	-
Amazon AI (Shanghai)	Amazon AI (Shanghai)	China	✓	-	✓	-
NILC	University of São Paulo	Brazil	✓	-	-	-
NUIG-DSI	National University of Ireland	Ireland	✓	-	-	-
CycleGT	Amazon	China	✓	-	✓	-
OSU Neural NLG	The Ohio State University	US	✓	✓	-	-
FBConvAI	Facebook	US	✓	✓	-	-

Table 4: WebNLG+ 2020 Participants.

with data augmentation. For data augmentation, they use the Baidu SKE dataset (194,747 RDF/Chinese text pairs) and automatically translate its text part into Russian.

ORANGE-NLG. Montella et al. (2020) explore data augmentation for RDF-to-English generation. They pre-train BART (Lewis et al., 2020) first on a corpus of Wikipedia sentences (57 million sentences) and second on a noisy RDF/English text corpus they created using Open Information Extraction on the collected sentences. For fine-tuning, they experiment with curriculum learning based on the size (number of triples) of the input. They find that pre-training and data augmentation does help improve results. Conversely, they found that curriculum learning leads to a drop in performance.

TGen. Kertkeidkachorn and Takamura (2020) introduce a pipeline model which first orders the input triples (plan selection) and second verbalises the resulting sequence of triples (verbalisation). Verbalisation is done using the T5 transformer-based encoder-decoder model (Raffel et al., 2020) trained through an unsupervised multi-tasking (span masking) on the Colossal Clean Crawled Corpus (C4) and fine-tuning on the RDF-to-English dataset. The Plan Selection model is learned using a ranking loss on a corpus which aligns each set of RDF triples with its possible linearisations and the corresponding texts (using the verbaliser) and where the plan which yields the text with the highest BLEU score is labelled as correct.

UPC-POE. Domingo Roig et al. (2020) attempt a semi-supervised, back translation approach where additional text data is retrieved from Wikipedia pages that are about entities similar to those present in the WebNLG+ dataset (using Wikipedia2vec embeddings for entities and words from Wikipedia). They then apply syntactic parsing to this additional text and integrate this synthetic data with the WebNLG+ data for training. The full dataset has around 350K instances. The model is a Transformer-based encoder-decoder with a BPE vocabulary of 7K subwords.

NILC. Sobrevilla Cabezudo and Salgueiro Pardo (2020) use the large BART Transformer Encoder-Decoder model and fine-tune it on the WebNLG+ data. The results are lower than the WebNLG+ baseline but preliminary investigations suggests that BART sometimes generates correct paraphrases for the reference.

NUIG-DSI. Pasricha et al. (2020) leverage the T5 transformer-based encoder-decoder model which was pre-trained on multiple supervised and unsupervised tasks. Before fine-tuning on the WebNLG+ data, they further pre-train T5 using a Mask Language Modelling objective (with 15% of the tokens masked) on two additional datasets: the WebNLG corpus and a corpus of DBpedia abstracts which consists of all abstracts for the entities which are present in the WebNLG+ training set.

3.3 Mono-task, Bilingual Approaches

cuni-ufal. The mBART model (Liu et al., 2020) is pre-trained for multilingual denoising on the large-scale multilingual CC25 corpus extracted from Common Crawl, which contains data in 25 languages. The noise function of mBART replaces

text spans of arbitrary length with a mask token (35% of the words in each instance) and permutes the order of sentences. To generate into both English and Russian, Kasner and Dusek (2020) fine-tune two separate mBART models for English and Russian on the WebNLG+ RDF-to-English and RDF-to-Russian datasets.

Huawei Noah's Ark Lab. Delexicalisation is used to help handle rare entities. Named entities are replaced by placeholders in the input and the output, the model is trained on the delexicalised data and the predictions are relexicalised before evaluation. While previous work on delexicalisation is mostly string based, Zhou and Lampouras (2020) propose a novel approach to delexicalisation which is based on embedding (semantic) similarity. To handle both English and Russian, they use LASER cross-lingual embeddings. To account for contextual variations, they complement the relexicalisation step with a contextualised post-editing model. They also explore the respective performance of delexicalisation, subwords and an approach combining both (using delexicalisation for unseen entities and word pieces for seen input).

OSU Neural NLG. Xintong et al. (2020) use the monolingual T5 model for English and the multilingual mBART model for Russian. Both models are fine-tuned on the WebNLG+ data. The authors also explore the impact of a reverse model reranking to rerank the model predictions after beam search.

FBConvAI. Yang et al. (2020) use BART for pre-training and explore different ways of modeling the RDF graph and its relation to natural language text. Different linearisation strategies (deph-first, bread-first traversal, bracketed representations) are compared. Multi-tasking and pipeline architectures are also examined to analyse how different ways of integrating generation with document planning (triples order) impact performance. To help bridge the gap between the input graph and the output linear structure, a second phase of pre-training is applied using DocRED, a noisy parallel corpus of sentences and their automatically extracted relation (17K entries). Lexicalisation of RDF properties are also curated from the WebNLG+ and the DocRED datasets.

3.4 Bi-Directional, Monolingual Approaches

Amazon AI (Shanghai). Zhao et al. (2020) introduced a two-step model for RDF-to-Text generation which combines a planner trained to learn the order in which triples should be verbalised and a decoder for verbalising each triple. Guo et al. (2020a) train Zhao et al. (2020)'s planner on the WebNLG+ dataset and use the pre-trained T5-large model to verbalise the linearised triples. For the Text-to-RDF task, entity linking is applied to the text and DBpedia is queried to retrieve the corresponding triples.

CycleGT. Guo et al. (2020b) present a weakly supervised method where generation and semantic parsing models are learned by bootstrapping from purely text and purely RDF data and iteratively mapping between the two forms. The T5 pre-trained sequence-to-sequence model is used to bootstrap the generation model. For semantic parsing, the authors use Qi et al. (2020) entity extraction model to identify all entities present in the input text and a multi-label classifier to predict the relation between pairs of entities. Each input text and each input graph is aligned with its back-translated version and the resulting aligned data for training. The two models are improved by repeatedly alternating the optimisation of each model. The text and the RDF data used to bootstrap the model are the WebNLG+ 2020 dataset, shuffled to ensure that the data is fully non parallel (text and RDF in each of the datasets are not aligned).

3.5 Bi-directional, Bi-lingual Approaches

bt5. Agarwal et al. (2020) use T5 as a pre-trained model and explores multilingual multi-task learning during pre-training and fine-tuning. For pre-training, their best model is T5 pre-trained on English and Russian Wikipedia and further trained on WMT English/Russian parallel corpus. For fine-tuning, they compare monolingual models, bilingual models multi-tasked on both languages and then fine-tuned for one and the same bilingual models fine tuned on a corpus derived from the WebNLG+ data by aligning English and Russian sentences and entities. They find that the later model provides significant improvements on unseen relations.

4 Evaluation Methodology

4.1 RDF-to-Text (Generation)

Automatic Metrics. The participating systems were automatically evaluated with some of the most popular traditional and novel text generation met-

rics. In the former group, we compared the textual outputs of the participating systems with their corresponding gold-standards using BLEU (Papineni et al., 2002), regular and with the Smoothing Function 3 proposed in (Chen and Cherry, 2014) (e.g., BLEU NLTK); METEOR (Lavie and Agarwal, 2007); TER (Snover et al., 2006) and chrF++ (Popović, 2017) (with word bigrams, character 6-grams and $\beta = 2$). Regarding the novel metrics, we computed BERTScore (Zhang et al., 2020) for English and Russian outputs and BLEURT (Sellam et al., 2020) (with `bleurt-base-128` version) for the English ones. The main difference between traditional and novel metrics is that the former measures the similarity between hypothesis and references using a discrete representation of their tokens, whereas the latter methods use a vector representation of these units. As an outcome of this shared task, we aim to investigate which one out these two kinds better correlate with human ratings.

For both considered languages, the participating systems were automatically evaluated in a multi-reference scenario. Each English hypothesis was compared with a maximum of 5 references, and each Russian one with a maximum of 7 references. On average, English data has 2.89 references per test instance, and Russian data has 2.52 references per instance. We requested the participants to provide their hypothesis in the detokenised and truecased form. Thus, the metrics were computed over the truecased format of the inputs. For the traditional metrics (e.g., BLEU, METEOR, chrF++, etc.), we tokenised the texts using the NLTK framework (Loper and Bird, 2002) for English, and `razdel`[2] for Russian. Novel metrics as BERTScore and BLEURT provide their own tokenisers.

Human Evaluation. We have conducted a human evaluation of all submitted systems for the RDF-to-Text task for both English and Russian data. In case of multiple submissions per participant for a single task, we asked to indicate the primary submission for human evaluation. Thus, we had **fourteen** submissions for English data and **six** submissions for Russian data. We have also evaluated baseline outputs and ground-truth references of both English and Russian data.

For both English and Russian data, we sampled 10% of RDF-text pairs from the respective test set for human evaluation in a random stratified fashion. Specifically, we sampled 178 triples from the English test set and 110 triples from the Russian test set. As Table 5 shows, we randomly chose samples based on the number of triples in a single data item. We also controlled for the type of the triples: our English data for human evaluation contained 54/37/87 samples for seen categories, unseen entities and unseen categories respectively. Russian data had triples of the first type only (seen categories). For each sample, we collected judgements from three different annotators. Our human annotators were recruited through the crowd-sourcing platform Amazon Mechanical Turk (MTurk) for English data and Yandex.Toloka for Russian data. They were asked to evaluate each sample based on the following criteria:

	Triple Set Size							
	1	*2*	*3*	*4*	*5*	*6*	*7*	*All*
English	36	40	30	31	22	9	10	178
Russian	26	20	19	20	22	0	3	110

Table 5: The number of samples per triple set size from the test set for both languages.

1. **Data Coverage**: Does the text include descriptions of all predicates presented in the data?

2. **Relevance**: Does the text describe only such predicates (with related subjects and objects), which are found in the data?

3. **Correctness**: When describing predicates which are found in the data, does the text mention correct the objects and adequately introduces the subject for this specific predicate?

4. **Text Structure**: Is the text grammatical, well-structured, written in acceptable English language?

5. **Fluency**: Is it possible to say that the text progresses naturally, forms a coherent whole and it is easy to understand the text?

Example tasks presented to the annotators (with criteria descriptions and examples of the data) are shown in the Appendix A for English and Appendix B for Russian. As can be seen from these examples, each annotator saw the following elements

[2] https://github.com/natasha/razdel

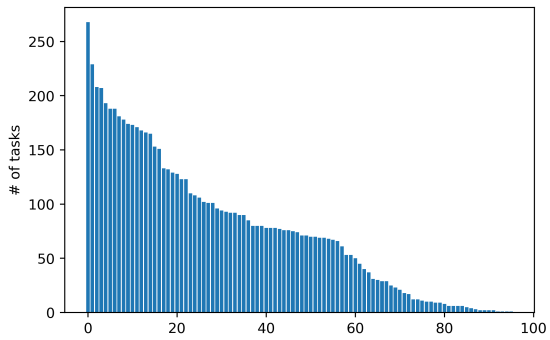
when working on our task: *(i)* the set of instructions with descriptions of each criterion, *(ii)* data (a collection of RDF triples), *(iii)* a system output (a text). Under each criterion description, a slider for the scale from 0 to 100 was given. Human annotators were required to use the slider and the scale to indicate the extent to which they agree with the statement about the specific measure. Each annotator was presented with a single evaluation sample per page. The full set of instructions is available in the GitHub challenge evaluation repository[3].

Our English tasks were available for annotators from English-speaking countries (the US, the UK, Australia, Ireland, Canada), who have completed more than 5,000 tasks on MTurk and had the approval rate of at least 95%. If a sample had 1, 2 or 3 RDF triples, we paid 0.15$ for the annotation of that sample. For triples of other sizes (4-7), we paid 0.20$ due to the higher task complexity. Our Russian tasks were available for the Russian-speaking annotators from Russia, Ukraine, Belarus and Kazakhstan. We paid the same amount of money for completing the Russian data annotation task as for the English data collection.

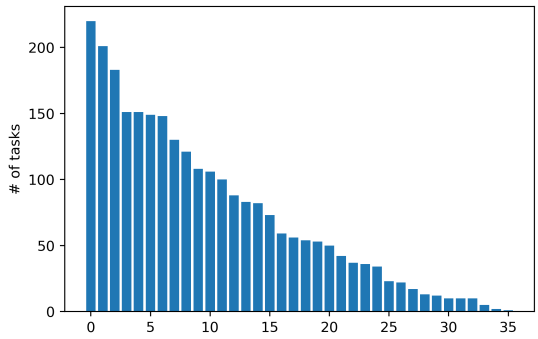
To ensure the quality in annotators' judgements, we conducted a round of *qualification tasks*. Only workers who have completed these tasks were allowed to participate in our primary tasks. The qualification tasks were created manually and included two examples of RDF-text pairs per single task. These tasks contained multiple instances of several types:

- The text correctly depicts and describes all information from the data (expected rating: high for all criteria).
- The text does not meet requirements for a single criterion (expected rating: low for the specific criterion).
- The text has many flaws across the majority of criteria (expected rating: low for most of the criteria).

A single annotator was qualified to work on the actual tasks if, given the results of qualification round *(i)* both qualification samples were evaluated as expected, *(ii)* evaluation of one qualification sample was slightly varied from what is expected. In all other cases, the annotator was not given access to our tasks. We also removed all annotators who were rating English ground-truth texts with low scores across multiple criteria. For Russian data, we manually controlled for this aspect since not all ground-truth texts are of high quality.



(a) English data: **MTurk**



(b) Russian data: **Yandex.Toloka**

Figure 1: Annotator statistics: # of annotators vs. # of tasks submitted per annotator.

We conducted two rounds of human evaluation for English data and have recruited **109** annotators. We have also imposed soft limitations on the number of samples an annotator is allowed to evaluate. In the first round, we allowed each worker to complete 150-170 tasks. In the second round, the range was changed to 130-150 tasks per annotator. Annotators from the first round (experienced annotators) were asked to participate in the second round. With this, we aimed at using their high level of expertise in our task to get better and more consistent judgements. For English data, we collected judgements from 109 annotators with 63 experienced annotators. Similarly, for Russian data, we recruited **37** annotators and each of them was allowed to submit 80-100 tasks in the first round and 120-140 tasks in the second round. We note that we softly

[3] https://github.com/WebNLG/GenerationEval/tree/humaneval/human_evaluation/en/hit_properties

controlled the number of possible task submissions per worker. We tracked the number of submitted tasks from each worker and restricted their access when the number had exceeded the limit. This update was performed every 5 minutes, and during this period, the worker could have submitted more tasks than allowed. Therefore, we do not set the maximally allowed number of submitted tasks per worker to a single number, but to a range of numbers instead. Fig. 1 demonstrates the distribution of task submissions for all our annotators.

Also, we manually checked submissions from each annotator who participated in our tasks. We have noticed the following patterns in the behaviour of bad annotators: first, their submissions contained identical scores (e.g., all 0s, all 100s, all 50s, etc.) for all criteria across all RDF-text sets. Second, their scores for several criteria were not logically correct (e.g., a low score for Data Coverage given that the text covers all predicates from the data). Third, bad submissions were typically sent in a short amount of time (around 10-20 seconds to complete a single task), and all bad annotators were highly active in submitting many tasks. Based on these patterns, we manually judged workers as either spammers or not. For English data, we identified 21 bad annotators, submitting around 25% of all data for English evaluation. We recollected 20% of these annotations, ensuring that their quality is reliable, and removed the other 5% of data from the results. For Russian data, we identified four malicious annotators who submitted around 5% of all data. We recollected these judgements.[4] Overall, we spent around 490 US dollars and 2,400 US dollars for human evaluation of Russian and English data, respectively.

Once the human evaluation was done, we preprocessed the ratings before computing the final human evaluation rankings for the systems:

- To diminish the differences between the scoring strategies of the distinct human raters, we normalized the scores of each participant by computing their z-scores (scores subtracted by the participant's overall mean score divided by their overall standard deviation).
- The standardised scores were averaged for each instance (as around 3 judgements were collected per instance), and then they were averaged across all sample instances (avg. z).
- We performed the Wilcoxon Rank-Sum Test to evaluate whether there is a statistically significant difference between the average evaluation scores of the systems. The result is shown as a system's rank, which was set measuring the pair-wise statistical tests between the averaged z-score results of a top-performing systems with the results of each of its lower-performing ones.
- We computed final human evaluation results for *(i)* the whole set of sampled test set outputs per system, *(ii)* for outputs per each test set type (seen categories, unseen entities, unseen categories).

Baselines. We used the FORGe generator (Mille et al., 2019a) as a baseline, an all-purpose grammar- and template-based generator that takes predicate-argument structures as input. FORGe was adapted to triple-based inputs such as the E2E and several DBpedia-oriented datasets — including WebNLG and WebNLG+ — with the addition of a module for the mapping of RDF to predicate-argument (external module) and a module for aggregation. It consists of 16 graph-transduction grammars that perform the following tasks as a pipeline: *(i)* aggregation of predicate-argument templates, *(ii)* definition of sentence structure for each resulting aggregated graph, *(iii)* introduction of idiosyncratic words and syntactic relations, *(iv)* syntax-based sentence aggregation and referring expression generation, and *(v)* linearisation and inflection. The grammars currently contain about 2,000 active rules, most of which are language- and domain-independent.[5] For instance, the micro-planning grammars use features such as the presence of repeated substructures to package some triples together, but do not target specific elements. Similarly, the sentence structures are chosen by default according to the configuration of the packaged semantic graph.

For the adaptation of the generator to the WebNLG+ dataset, the following steps were required: *(i)* for each individual property, one predicate-argument template (in a PropBank-like fashion (Babko-Malaya, 2005)) was handcrafted, *(ii)* for each lexical unit used in the templates, a

[4] The results of the human evaluation in this report are the final results. Please note that the system description papers might report/analyse **non-filtered results** (e.g. human evaluation results based on the annotators' data which has not been manually inspected), if not stated otherwise.

[5] 4 rules have been specifically designed to cope with some particular WebNLG and WebNLG+ inputs.

lexicon entry was added with the description of its subcategorisation pattern and minimal collocation information, *(iii)* each inflected form needed for the verbalisation was added to morphological dictionary, and *(iv)* the coverage of a few rules was extended to handle new configurations. Most of the templates, lexical units and full-fledged forms had already been established for the first edition of the WebNLG challenge. The training and development sets were used to see how the different properties are verbalised and to get inspiration for crafting the predicate-argument templates; basic templates were also added for each unseen property.

During the mapping from the RDF triples to the predicate-argument structures, we added information (in the form of feature-value pairs) obtained by querying DBpedia (class, number, cardinality, gender), removed all Wikipedia disambiguation information, i.e. what is in parentheses in the subject and object values, added generic processing rules to normalise numbers and dates, and split comma-separated object values. We also added rules specific to the WebNLG+ dataset to handle semantically overlapping triples (e.g. the triple about a person being deceased was removed when there was a triple about the death date). Before being sent to FORGe, the triples were ordered in a way that the ones with common subject and/or object are consecutive, the triples involving the most frequent entities being placed at the beginning. Since the semantic and syntactic aggregation grammars only group a triple/syntactic subtree with a (directly or indirectly) preceding element, this partially establishes the final order in which the triples are verbalised.

For English, the second baseline is the FORGe system as submitted at the WebNLG 2017 task (Mille et al., 2019b), which we run using the WebNLG+ predicate-argument templates, lexical and morphological resources. The second baseline does not have access to the improvements in terms of grammars (in particular about sentence structuring and triple- and syntax-based aggregations) that the first baseline has. For Russian, we generated English texts using the first baseline, and translated the outputs using Google Translate.

The motivation behind using a rule-based baseline is to be found in the 2017 task, in which FORGe got stable results in the human evaluations for the seen and unseen categories, with high scores according to all evaluation criteria. We expect the baseline to score reasonably high in terms of human assessments, in particular according to coverage, correctness and relevance, since there are no hallucinations in grammar-based systems, and we ensured that all the properties are covered. For fluency and text structure, we expect the scores to be lower, but to still provide a strong baseline.

4.2 Text-to-RDF (Semantic Parsing)

For the Semantic Parsing task, we did not conduct the human evaluation of the submitted systems. Thus, only automatic metrics were used to evaluate the performance.

Automatic Metrics. Precision, Recall, and F1 score metrics were calculated for the Text-to-RDF task. This calculation was based on the Named Entity Evaluation for SemEval 2013, Task 9.1 (Segura-Bedmar et al., 2013). First, the triples were pre-processed: the snake cased subject and object, and camel cased predicate, were converted to regular strings. Then, the resulting strings were lowercased, quotation marks were removed, and if an object contained text between parentheses (e.g., "The Honeymoon Killers (American band)"), that was removed as well. After this pre-processing step, the evaluation script looked for the optimal alignment between each candidate and reference triple. Then, generated triples and gold standard triples were converted separately to a string with start and end index information. Furthermore, the subject, verb, and object information were saved as an entity for the evaluation. With this information, using metrics based on Named Entity Evaluation becomes possible. Four different ways to measure Precision, Recall, and F1 score were investigated (see also Table 7):[6]

1. **Strict**: Exact match of the candidate triple element with the reference triple element is required. And the element type (subject, predicate, object) should match with the reference.

2. **Exact**: Exact match of the candidate triple element with the reference triple element is required, and the element type (subject, predicate, object) is irrelevant.

3. **Partial**: The candidate triple element should match at least partially with the reference triple element, and the element type (subject, predicate, object) is irrelevant.

[6] See Batista (2018) for a more detailed explanation of the different measures.

Team Name	BLEU	BLEU NLTK	METEOR	chrF++	TER	BERT Precision	BERT Recall	BERT F1	BLEURT
Amazon AI (Shanghai)	0.540	0.535	0.417	0.690	0.406	0.960	0.957	0.958	0.620
OSU Neural NLG	0.535	0.532	0.414	0.688	0.416	0.958	0.955	0.956	0.610
FBConvAI *	0.527	0.523	0.413	0.686	0.423	0.957	0.955	0.956	0.600
bt5	0.517	0.517	0.411	0.679	0.435	0.955	0.954	0.954	0.600
NUIG-DSI	0.517	0.514	0.403	0.669	0.417	0.959	0.954	0.956	0.610
cuni-ufal	0.503	0.500	0.398	0.666	0.435	0.954	0.950	0.951	0.570
DANGNT-SGU	0.407	0.405	0.393	0.646	0.511	0.940	0.946	0.943	0.450
CycleGT	0.446	0.432	0.387	0.637	0.479	0.949	0.949	0.948	0.540
RALI-Université de Montréal	0.403	0.393	0.386	0.634	0.504	0.944	0.944	0.944	0.450
TGen	0.509	0.482	0.384	0.636	0.454	0.952	0.947	0.949	0.540
Baseline-FORGE2020	0.406	0.396	0.373	0.621	0.517	0.946	0.941	0.943	0.470
Huawei Noah's Ark Lab	0.396	0.387	0.372	0.613	0.536	0.935	0.937	0.935	0.370
Baseline-FORGE2017	0.379	0.371	0.364	0.606	0.553	0.933	0.935	0.930	0.420
NILC	0.320	0.313	0.350	0.545	0.629	0.920	0.922	0.920	0.400
ORANGE-NLG	0.384	0.377	0.343	0.584	0.587	0.927	0.922	0.924	0.330
UPC-POE	0.391	0.379	0.337	0.579	0.564	0.933	0.927	0.929	0.370

Table 6: Automatic Evaluation results for **English** RDF-to-text system submissions **on the full test set**. The teams are sorted by METEOR scores. Two baseline systems (from the previous and current WebNLG challenges) are coloured in grey. * indicates late submission.

4. **Type**: The candidate triple element should match at least partially with the reference triple element, and the element type (subject, predicate, object) should match with the reference.

Gold entity	Gold string	Pred entity	Pred string	Type	Partial	Exact	Strict
SUB	Bionico			MIS	MIS	MIS	MIS
		OBJ	Granola	SPU	SPU	SPU	SPU
PRED	place	PRED	birth place	COR	PAR	INC	INC
SUB	Bionico	OBJ	Bionico	INC	COR	COR	INC
PRED	architect	PRED	architect	COR	COR	COR	COR
SUB	Capers	OBJ	Super Capers	INC	PAR	INC	INC

Table 7: Examples of possible error types for semantic parsing, and how these are interpreted by the measures. COR = correct, INC = incorrect, PAR = partial, MIS = missed, SPU = spurious.

For development purposes, the evaluation script also provided information about the number of correct, incorrect, partial missed, spurious, possible, and actual matches for the four measures.

Baselines. A baseline was constructed by using Stanford CoreNLP's Open Information Extraction module (Manning et al., 2014) on the texts in the test set. This module allows for the extraction of subjects, relations, and objects in a string without any training necessary. Extraction of these triples was limited to 10 per text, to avoid memory overflow errors when running the evaluation script. As this Open Information Extraction module was only developed for English, the Russian sentences were translated to English using DeepL,[7] before extracting the RDF triples using Stanford CoreNLP's Open Information Extraction module.

5 Results of Automatic Evaluation

In this section, we present the automatic scores on English and Russian datasets for both tasks, namely, RDF-to-text and Text-to-RDF. For English, we discuss the automatic scores, and make a distinction between results on *(i)* the entire dataset, *(ii)* seen semantic categories, *(iii)* seen semantic categories but unseen entities and *(iv)* unseen semantic categories. For Russian, the only reported results are for the entire dataset, as the test set only contained seen entities and categories.

5.1 RDF-to-text

English. Table 6 displays the results of the automatic evaluation of the RDF-to-text systems, ordered by METEOR scores on the entire test set. Most systems (10 out of 15) outperformed at least one of the baselines, which are highlighted in gray.

Following a popular trend in Natural Language Processing, fine-tuning large pre-trained language models, such as BART and T5, was a common strategy among the participants to achieve better results. From the 6 best ranked systems for instance, 4 made use of T5 (1st, 2nd, 4th and 5th), the third used BART whereas the sixth generates the texts using a multilingual version of the latter called mBART.

[7] https://www.deepl.com/en/translator

Seen Categories

Team Name	BLEU	BLEU NLTK	METEOR	chrF++	TER	BERT Precision	BERT Recall	BERT F1	BLEURT
FBConvAI *	0.613	0.608	0.436	0.730	0.395	0.964	0.961	0.962	0.610
OSU Neural NLG	0.612	0.607	0.434	0.727	0.393	0.964	0.960	0.962	0.610
Amazon AI (Shanghai)	0.604	0.596	0.434	0.723	0.404	0.964	0.961	0.962	0.590
bt5	0.611	0.611	0.433	0.725	0.391	0.965	0.961	0.963	0.600
ORANGE-NLG	0.593	0.584	0.428	0.712	0.415	0.963	0.957	0.960	0.600
cuni-ufal	0.591	0.588	0.422	0.712	0.403	0.964	0.957	0.960	0.580
NUIG-DSI	0.583	0.579	0.416	0.699	0.408	0.964	0.958	0.960	0.600
NILC	0.562	0.550	0.409	0.700	0.430	0.961	0.957	0.958	0.580
DANGNT-SGU	0.463	0.452	0.406	0.675	0.523	0.944	0.949	0.946	0.420
Huawei Noah's Ark Lab	0.497	0.482	0.402	0.674	0.504	0.950	0.949	0.949	0.460
CycleGT	0.474	0.458	0.394	0.654	0.490	0.951	0.950	0.950	0.500
RALI-Université de Montréal	0.437	0.417	0.394	0.652	0.530	0.947	0.949	0.948	0.420
Baseline-FORGE2020	0.430	0.415	0.387	0.650	0.563	0.945	0.942	0.943	0.410
Baseline-FORGE2017	0.412	0.398	0.384	0.642	0.599	0.938	0.938	0.936	0.330
TGen	0.610	0.518	0.381	0.641	0.432	0.961	0.948	0.954	0.550
UPC-POE	0.512	0.495	0.373	0.648	0.478	0.957	0.943	0.949	0.500

Unseen Entities

Team Name	BLEU	BLEU NLTK	METEOR	chrF++	TER	BERT Precision	BERT Recall	BERT F1	BLEURT
OSU Neural NLG	0.524	0.520	0.416	0.694	0.398	0.963	0.960	0.961	0.650
NUIG-DSI	0.528	0.523	0.415	0.691	0.381	0.964	0.961	0.962	0.670
bt5	0.508	0.505	0.415	0.687	0.411	0.961	0.959	0.959	0.650
FBConvAI *	0.503	0.497	0.414	0.689	0.411	0.962	0.961	0.961	0.650
Amazon AI (Shanghai)	0.523	0.517	0.413	0.691	0.394	0.963	0.960	0.961	0.650
cuni-ufal	0.512	0.500	0.406	0.687	0.417	0.960	0.958	0.959	0.630
TGen	0.507	0.504	0.405	0.672	0.410	0.958	0.956	0.957	0.610
RALI-Université de Montréal	0.435	0.422	0.395	0.658	0.464	0.950	0.950	0.949	0.530
DANGNT-SGU	0.411	0.409	0.391	0.655	0.493	0.947	0.952	0.949	0.520
CycleGT	0.466	0.445	0.390	0.653	0.448	0.956	0.954	0.955	0.600
Baseline-FORGE2020	0.402	0.393	0.384	0.648	0.476	0.949	0.950	0.949	0.550
Huawei Noah's Ark Lab	0.424	0.411	0.375	0.631	0.487	0.944	0.944	0.944	0.480
Baseline-FORGE2017	0.381	0.366	0.367	0.626	0.515	0.933	0.941	0.932	0.500
NILC	0.219	0.215	0.340	0.509	0.671	0.914	0.919	0.916	0.420
ORANGE-NLG	0.358	0.354	0.326	0.565	0.590	0.929	0.913	0.920	0.260
UPC-POE	0.353	0.338	0.324	0.569	0.570	0.937	0.929	0.932	0.400

Unseen Categories

Team Name	BLEU	BLEU NLTK	METEOR	chrF++	TER	BERT Precision	BERT Recall	BERT F1	BLEURT
Amazon AI (Shanghai)	0.492	0.491	0.404	0.660	0.413	0.957	0.953	0.954	0.600
OSU Neural NLG	0.474	0.474	0.397	0.652	0.437	0.953	0.951	0.951	0.570
FBConvAI *	0.462	0.463	0.394	0.647	0.444	0.951	0.949	0.950	0.570
bt5	0.440	0.441	0.393	0.636	0.470	0.948	0.948	0.947	0.560
NUIG-DSI	0.456	0.454	0.388	0.632	0.438	0.953	0.949	0.950	0.580
DANGNT-SGU	0.359	0.364	0.384	0.617	0.512	0.935	0.942	0.938	0.420
CycleGT	0.409	0.405	0.379	0.615	0.486	0.945	0.946	0.945	0.540
TGen	0.438	0.436	0.379	0.618	0.472	0.947	0.944	0.945	0.500
RALI-Université de Montréal	0.359	0.360	0.375	0.606	0.507	0.940	0.939	0.939	0.420
cuni-ufal	0.422	0.425	0.375	0.617	0.460	0.946	0.942	0.943	0.520
Baseline-FORGE2020	0.376	0.370	0.357	0.584	0.510	0.944	0.936	0.940	0.440
Baseline-FORGE2017	0.346	0.343	0.347	0.565	0.544	0.930	0.930	0.925	0.390
Huawei Noah's Ark Lab	0.291	0.293	0.345	0.553	0.575	0.922	0.926	0.924	0.230
UPC-POE	0.295	0.288	0.316	0.526	0.608	0.918	0.917	0.917	0.180
NILC	0.162	0.161	0.311	0.435	0.719	0.900	0.905	0.902	0.190
ORANGE-NLG	0.233	0.229	0.288	0.485	0.680	0.907	0.906	0.906	0.030

Table 8: Automatic Evaluation results for the RDF-to-text task for English on seen categories, unseen entities, and unseen categories. * indicates late submission.

Team Name	BLEU	BLEU NLTK	METEOR	chrF++	TER	BERT Precision	BERT Recall	BERT F1
bt5	0.516	0.521	0.676	0.683	0.420	0.909	0.907	0.907
cuni-ufal	0.529	0.532	0.672	0.677	0.398	0.914	0.905	0.909
Huawei Noah's Ark Lab	0.468	0.468	0.632	0.637	0.456	0.899	0.890	0.893
FBConvAI *	0.453	0.451	0.617	0.641	0.452	0.903	0.894	0.898
OSU Neural NLG	0.473	0.477	0.616	0.622	0.453	0.897	0.882	0.888
med	0.431	0.430	0.576	0.595	0.487	0.898	0.873	0.884
Baseline-FORGE2020	0.255	0.256	0.467	0.514	0.665	0.841	0.835	0.837

Table 9: Automatic Evaluation results for **Russian** RDF-to-text system submissions **on the full test set**. The systems are sorted by METEOR score and the baseline system is coloured in grey. * indicates late submission.

Team Name	Match	F1	Precision	Recall
Amazon AI (Shanghai)	Exact	0.689	0.689	0.690
Amazon AI (Shanghai)	Ent_Type	0.700	0.699	0.701
Amazon AI (Shanghai)	Partial	0.696	0.696	0.698
Amazon AI (Shanghai)	Strict	0.686	0.686	0.687
bt5	Exact	0.682	0.670	0.701
bt5	Ent_Type	0.737	0.721	0.762
bt5	Partial	0.713	0.700	0.736
bt5	Strict	0.675	0.663	0.695
CycleGT	Exact	0.342	0.338	0.349
CycleGT	Ent_Type	0.343	0.335	0.356
CycleGT	Partial	0.360	0.355	0.372
CycleGT	Strict	0.309	0.306	0.315
Baseline	Exact	0.158	0.154	0.164
Baseline	Ent_Type	0.193	0.187	0.202
Baseline	Partial	0.200	0.194	0.211
Baseline	Strict	0.127	0.125	0.130

(a) **English** submissions

Team Name	Match	F1	Precision	Recall
bt5	Exact	0.911	0.907	0.917
bt5	Ent_Type	0.923	0.918	0.930
bt5	Partial	0.917	0.913	0.924
bt5	Strict	0.911	0.907	0.917
Baseline	Exact	0.119	0.113	0.129
Baseline	Ent_Type	0.147	0.137	0.163
Baseline	Partial	0.156	0.146	0.172
Baseline	Strict	0.087	0.083	0.094

(b) **Russian** submissions

Table 10: Automatic Evaluation results for Text-to-RDF (Semantic Parsing) task **for both languages** represented with Macro scores.

In the comparison between rule-based and neural approaches, results of the latter were usually higher than the former. Out of the 4 rule-based systems (including the baselines), `DANGNT-SGU` was the one that ranked highest in the automatic evaluation, being in the 7th position.

Table 8 depicts the results distinguished by *(i)* trials from semantic categories seen during training, *(ii)* trials from seen categories but with entities that were unseen during training and *(iii)* trials from categories fully unseen during training. We hypothesise that the difficulty increases along the different test sets: converting RDFs from semantic categories seen during training to texts is easier than generating from unseen semantic categories, where we want to evaluate how well the models can generalize. This hypothesis is indeed supported when looking at the number of systems that outperform the baselines (mostly based on their METEOR scores). 12 out of 15 systems were better than both baselines in the seen categories, whereas only 10 outperform the baselines in the fully unseen categories.

Across the three kinds of evaluation, the top performing systems were basically the same, except for `NUIG-DSI` which took the second position in the unseen entities, showing a better generalisation capability in comparison to the other top 5 systems. Conversely, `ORANGE-NLG` and `NILC` were ranked better when generating text for RDFs seen during training, but performed poorly when information not seen during training was present in the input RDF triples.

Russian. Table 9 shows the automatic evaluation results for the entire test set. Out of the 6 RDF-to-text systems for Russian, 5 are bilingual and were also submitted for the English variation of the task. These system were also the top performing ones, ranked higher than the unique monolingual systems for Russian. From the bilingual approaches, 2 are based on a T5 fine-tuned language model and 2 on the BART language model. In the comparison among them, `bt5` showed a superior performance in terms of METEOR and chrF++, however, on the other metrics `cuni-ufal` performed better. Our baseline was the system which had the lowest scores. Such results were expected due to its out-

Team Name	Match	F1	Precision	Recall
Seen Categories				
bt5	Exact	0.877	0.875	0.880
bt5	Ent_Type	0.888	0.885	0.891
bt5	Partial	0.883	0.881	0.886
bt5	Strict	0.877	0.875	0.880
Amazon AI (Shanghai)	Exact	0.693	0.693	0.694
Amazon AI (Shanghai)	Ent_Type	0.718	0.718	0.718
Amazon AI (Shanghai)	Partial	0.707	0.707	0.707
Amazon AI (Shanghai)	Strict	0.693	0.692	0.693
CycleGT	Exact	0.548	0.541	0.560
CycleGT	Ent_Type	0.618	0.599	0.648
CycleGT	Partial	0.585	0.572	0.607
CycleGT	Strict	0.545	0.538	0.558
Baseline	Exact	0.165	0.163	0.170
Baseline	Ent_Type	0.211	0.205	0.221
Baseline	Partial	0.211	0.205	0.221
Baseline	Strict	0.140	0.139	0.143
Unseen Categories				
Amazon AI (Shanghai)	Exact	0.658	0.657	0.660
Amazon AI (Shanghai)	Ent_Type	0.661	0.660	0.663
Amazon AI (Shanghai)	Partial	0.662	0.661	0.663
Amazon AI (Shanghai)	Strict	0.655	0.655	0.657
bt5	Exact	0.551	0.540	0.568
bt5	Ent_Type	0.653	0.636	0.679
bt5	Partial	0.609	0.595	0.631
bt5	Strict	0.539	0.528	0.555
CycleGT	Exact	0.223	0.222	0.227
CycleGT	Ent_Type	0.195	0.193	0.200
CycleGT	Partial	0.234	0.233	0.240
CycleGT	Strict	0.181	0.179	0.183
Baseline	Exact	0.140	0.137	0.146
Baseline	Ent_Type	0.179	0.172	0.188
Baseline	Partial	0.188	0.182	0.199
Baseline	Strict	0.105	0.103	0.108
Unseen Entities				
Amazon AI (Shanghai)	Exact	0.746	0.746	0.747
Amazon AI (Shanghai)	Ent_Type	0.751	0.750	0.753
Amazon AI (Shanghai)	Partial	0.751	0.751	0.753
Amazon AI (Shanghai)	Strict	0.740	0.739	0.741
bt5	Exact	0.649	0.617	0.701
bt5	Ent_Type	0.675	0.640	0.731
bt5	Partial	0.664	0.631	0.718
bt5	Strict	0.645	0.614	0.697
CycleGT	Exact	0.239	0.238	0.247
CycleGT	Ent_Type	0.185	0.183	0.188
CycleGT	Partial	0.243	0.242	0.252
CycleGT	Strict	0.179	0.178	0.182
Baseline	Exact	0.184	0.178	0.194
Baseline	Ent_Type	0.196	0.190	0.205
Baseline	Partial	0.210	0.202	0.224
Baseline	Strict	0.152	0.149	0.157

Table 11: Text-to-RDF (Semantic Parsing) results per test data type for English: we show Macro scores for seen categories, unseen categories and unseen entities.

puts were obtained by simply automatically translating the texts generated by the English baseline.

5.2 Text-to-RDF

English. Table 10a shows the results for the entire test set. `bt5` and `Amazon AI (Shanghai)` achieved similar results. `bt5` achieved higher scores than `Amazon AI (Shanghai)` on the more liberal `Ent_Type` and `Partial` measurement types, while `Amazon AI` showed better results on the stricter Exact and Strict matches.

Table 11 depicts the results distinguished by (1) trials from categories seen during training, (2) trials from seen categories but with unseen entities during training and (3) trials from domains fully unseen during training. For the seen categories, `bt5` demonstrated superior performance, achieving higher F1 scores across all metrics in comparison to the other participants.

On unseen categories, `Amazon AI (Shanghai)` took the first place on this test set showing a generalisation capability for handling unseen categories. `Amazon AI (Shanghai)` achieved nearly 0.66 F1 across all metrics while the second-best performing system, `bt5`, achieved 0.60 on average. `CycleGT` performed slightly better than the baseline but still improved over the baseline results.

`Amazon AI (Shanghai)` also took first place on the unseen entities set and achieved nearly 0.75 F1 across all metrics while the second-best performing system, `bt5`, achieved 0.64 on average. Similar to the unseen categories, `CycleGT` also achieved a slight improvement over the baseline for the unseen entities. The results of all systems on this test set show a similar tendency as the unseen categories test set, but, overall, the scores were higher than on unseen categories. The results on the unseen test sets suggest that the `bt5` and `Amazon AI (Shanghai)` models were reasonably capable of generalising the position of the entities in the text. This is further corroborated by the relatively small drop of those models between the seen and unseen test sets. However, the differing nature of the entities across categories made generalisation more difficult.

Russian. Table 10b shows the results on the entire test set, which was comprised of seen categories only. `bt5` was the only system to perform this task and achieved impressive F1 results in comparison to the baseline on all metrics. It also achieved even higher scores than the `bt5` system obtained on the English seen categories test set.

	DATA COVERAGE			RELEVANCE			CORRECTNESS			TEXT STRUCTURE			FLUENCY		
TEAM NAME	**Rank**	**Avg. Z**	**Avg. Raw**	**Rank**	**Avg. Z**	**Avg. Raw**	**Rank**	**Avg. Z**	**Avg. Raw**	**Rank**	**Avg. Z**	**Avg. Raw**	**Rank**	**Avg. Z**	**Avg. Raw**
FBConvAI *	2	0.151	93.169	2	0.117	93.898	1	0.206	92.700	1	0.319	93.089	1	0.327	90.837
AmazonAI (Shanghai)	1	0.222	94.393	1	0.214	95.196	1	0.248	93.531	1	0.305	92.951	1	0.326	90.286
OSU Neural NLG	1	0.235	95.123	1	0.163	94.615	1	0.224	93.409	1	0.289	92.438	1	0.323	90.066
WebNLG-2020-REF	1	0.251	95.442	1	0.139	94.392	1	0.256	94.149	1	0.254	92.105	1	0.279	89.846
NUIG-DSI	2	0.116	92.063	1	0.161	94.061	1	0.189	92.053	1	0.258	91.588	2	0.233	88.898
bt5	2	0.161	93.836	1	0.184	95.220	1	0.224	93.583	1	0.236	91.914	2	0.218	88.688
cuni-ufal	2	0.155	93.291	1	0.164	94.555	1	0.161	91.587	1	0.208	90.752	2	0.185	87.642
TGen	3	-0.075	88.176	1	0.132	92.640	2	0.074	88.626	1	0.168	89.041	2	0.182	86.163
CycleGT	3	0.023	91.231	1	0.125	93.370	2	0.071	89.846	2	0.045	87.879	3	0.072	84.820
Baseline-FORGE2020	1	0.170	92.892	1	0.161	93.784	1	0.190	91.794	2	0.039	87.400	3	0.011	82.430
Baseline-FORGE2017	2	0.127	92.066	2	0.113	92.588	2	0.13	90.138	2	-0.064	85.737	4	-0.143	80.941
DANGNT-SGU	1	0.259	95.315	1	0.185	94.856	1	0.179	92.489	3	-0.203	83.501	4	-0.161	78.594
RALI-Université de Montréal	1	0.272	95.204	1	0.171	94.810	1	0.163	92.128	3	-0.285	81.835	4	-0.241	77.759
ORANGE-NLG	5	-0.554	79.959	4	-0.710	79.887	4	-0.668	74.977	3	-0.338	80.462	5	-0.332	75.675
Huawei Noah's Ark Lab	4	-0.310	84.743	3	-0.425	85.265	3	-0.389	80.760	3	-0.373	80.219	5	-0.369	75.205
NILC	4	-0.477	81.605	3	-0.499	83.522	3	-0.589	76.702	3	-0.402	80.463	5	-0.408	74.851
UPC-POE	6	-0.782	75.845	4	-0.531	82.051	4	-0.701	74.374	4	-0.456	78.503	5	-0.508	72.280

Table 12: Human Evaluation results for **English**: scores for **all** test data types. The systems are sorted by averaged Fluency raw scores. The colour intensity signifies final ranking in terms of averaged raw scores: more intense colour reflects higher performance for the specific criterion. * indicates late submission.

6 Results of Human Evaluation

In this section, we describe the results of the human evaluation conducted on a sample of the outputs of RDF-to-text system submissions for both English and Russian data. For English we evaluate systems' performance for *(i)* the full sampled test subset, *(ii)* subsets of each of the triple categories (seen categories, unseen entities, unseen categories). For Russian, we provide results for the full sampled test subset only. The final results for all test data types are shown in Table 12 for English systems. In Table 13 we evaluate systems for each of the test data types separately for the English data. Table 14 shows results for Russian system submissions.

English. Table 12 shows the results of the human evaluation for the RDF-to-text task for English system submissions. We first look at the differences between the results of the human and automatic evaluation: although the Fluency and Text Structure ratings of the rule-based systems (`RALI-Université-Montréal`, `DANGNT-SGU`, `Baseline-FORGE2020`) ranked similar to the automatic metrics in the lower part of the leaderboard, their human ratings for Data Coverage, Relevance and Correctness were among the highest.

Regarding Text Structure and Fluency, results of neural approaches as `FBConvAI`, `AmazonAI (Shanghai)` and `OSU Neural NLG` were rated surprisingly high, sharing the same ranking cluster with the ground-truth references (`WebNLG-2020-REF`). As expected, in terms of Data Coverage and Relevance, the rule-based participating systems (`RALI-Université-Montréal`, `DANGNT-SGU`) performed quite strongly, being in the same cluster as the references.

In fact, the relation between Fluency and Data Coverage is noticeably different: although systems based on fine-tuned T5 and BART language models ranked on the top for Fluency (`Amazon AI`, `FBConvAI`, `OSU Neural NLG`), the ones based on the latter language model (BART, `FBConvAI`) suffered a drop in performance in terms of Data Coverage, whereas the ones based on the former (T5) performed similarly to the rule-based approaches.

Table 13 depicts the human ratings for the RDF-to-text task in English, discriminated by three types of data: seen categories, unseen entities and unseen categories. Across the three different types of data, it is possible to notice some of the tendencies that were also found for the automatic evaluation. For instance, the difference in performance between the three kinds of data from models like `NILC` and `ORANGE-NLG`, which introduce good results for trials from semantic categories seen during training, but fail to generalise to new entities and semantic categories. For unseen categories which were not presented during training, most of the models were not able to outperform the ground-truth references (`WebNLG-2020-REF`) across multiple criteria.

Note that the scores for unseen categories are generally lower for all systems compared to the scores for seen categories and unseen entities. Overall, while rule-based systems (`RALI-Université-Montréal`, `DANGNT-SGU`) perform well for the criteria which evaluate connection between RDF triple and the text (Data Coverage, Relevance, Correctness), for the categories which evaluate naturalness and

	Data Coverage			Relevance			Correctness			Text Structure			Fluency		
Team Name	**Rank**	**Avg. Z**	**Avg. Raw**	**Rank**	**Avg. Z**	**Avg. Raw**	**Rank**	**Avg. Z**	**Avg. Raw**	**Rank**	**Avg. Z**	**Avg. Raw**	**Rank**	**Avg. Z**	**Avg. Raw**
FBConvAI *	1	0.178	93.543	2	0.112	93.111	1	0.261	93.472	1	0.326	92.966	1	0.358	91.654
bt5	1	0.196	94.460	1	0.222	95.167	1	0.312	94.843	1	0.264	91.846	1	0.280	89.892
cuni-ufal	1	0.257	94.941	1	0.203	94.870	1	0.273	93.886	1	0.263	91.429	1	0.281	89.454
OSU Neural NLG	1	0.176	94.287	2	0.084	93.373	1	0.233	94.015	1	0.239	91.599	1	0.253	88.651
NUIG-DSI	2	0.059	91.253	1	0.178	94.512	2	0.162	92.494	1	0.234	90.744	1	0.180	88.611
WebNLG-2020-REF	1	0.264	95.491	1	0.135	94.142	1	0.236	93.355	1	0.198	91.225	1	0.225	88.136
AmazonAI (Shanghai)	1	0.258	94.090	1	0.170	93.586	1	0.295	93.691	1	0.293	91.154	1	0.308	87.750
NILC	1	0.225	94.448	1	0.266	96.269	1	0.212	93.071	1	0.212	91.225	2	0.155	87.306
ORANGE-NLG	2	0.109	92.593	2	0.112	93.673	2	0.145	91.478	2	0.074	88.034	2	0.112	85.302
Huawei	2	0.101	92.173	2	0.011	92.222	2	0.080	90.269	2	0.067	88.380	2	0.064	85.111
CycleGT	3	-0.137	88.386	1	0.125	92.120	2	0.062	88.633	3	-0.121	84.262	2	-0.036	83.287
TGen	3	-0.394	81.670	2	0.074	91.099	2	-0.028	86.793	2	0.034	86.886	2	0.005	83.037
Baseline-FORGE2020	1	0.280	95.296	1	0.153	94.568	1	0.226	93.593	2	0.074	87.040	2	0.030	82.664
UPC-POE	4	-0.404	82.173	2	-0.019	90.503	3	-0.115	84.858	2	0.096	87.309	2	-0.077	80.577
DANGNT-SGU	1	0.239	94.367	1	0.164	93.596	2	0.140	90.772	3	-0.132	84.691	3	-0.159	79.559
RALI-Université de Montréal	1	0.274	93.846	1	0.148	93.049	2	0.198	91.423	3	-0.170	82.614	3	-0.157	79.238
Baseline-FORGE2017	2	0.065	90.253	2	-0.043	89.568	2	0.042	87.608	3	-0.160	82.892	3	-0.406	75.037

(a) **Seen categories**

	Data Coverage			Relevance			Correctness			Text Structure			Fluency		
Team Name	**Rank**	**Avg. Z**	**Avg. Raw**	**Rank**	**Avg. Z**	**Avg. Raw**	**Rank**	**Avg. Z**	**Avg. Raw**	**Rank**	**Avg. Z**	**Avg. Raw**	**Rank**	**Avg. Z**	**Avg. Raw**
AmazonAI (Shanghai)	1	0.291	95.532	1	0.272	96.329	1	0.293	94.703	1	0.348	94.288	1	0.452	93.365
TGen	1	0.209	93.649	1	0.234	95.356	1	0.230	91.883	1	0.347	92.347	1	0.448	91.869
FBConvAI *	2	0.139	93.536	2	0.169	95.644	1	0.201	94.000	1	0.334	94.405	1	0.365	91.599
OSU Neural NLG	1	0.158	94.203	1	0.253	95.662	1	0.178	92.338	1	0.292	92.482	1	0.385	91.293
WebNLG-2020-REF	1	0.283	95.991	1	0.315	97.117	1	0.268	95.171	1	0.281	93.189	1	0.285	90.788
bt5	1	0.158	93.734	2	0.146	95.351	1	0.154	93.239	1	0.263	91.766	1	0.318	89.595
NUIG-DSI	1	0.165	91.752	1	0.181	93.694	1	0.260	92.446	1	0.358	93.041	1	0.303	89.577
CycleGT	2	0.094	92.703	1	0.181	95.198	1	0.171	92.541	1	0.233	92.036	1	0.286	89.189
cuni-ufal	1	0.206	93.937	1	0.213	94.995	1	0.149	91.086	1	0.314	93.243	2	0.098	86.559
Baseline-FORGE2017	1	0.196	92.207	1	0.266	93.797	1	0.317	91.302	2	0.003	85.644	2	0.039	83.604
RALI-Université de Montréal	1	0.317	95.775	1	0.194	95.338	1	0.215	93.333	2	-0.088	86.559	2	-0.019	83.140
Baseline-FORGE2020	1	0.161	93.360	1	0.271	96.099	1	0.199	92.635	2	-0.011	88.243	2	0.025	82.126
Huawei Noah's Ark Lab	2	-0.259	85.041	3	-0.366	85.559	2	-0.242	84.126	2	-0.204	83.383	2	-0.221	79.315
DANGNT-SGU	1	0.230	95.329	1	0.249	96.658	1	0.160	93.459	2	-0.245	81.977	2	-0.116	78.599
ORANGE-NLG	3	-0.624	78.149	3	-0.697	78.950	3	-0.738	71.342	2	-0.285	80.505	3	-0.263	74.586
NILC	3	-0.343	84.896	3	-0.299	88.230	2	-0.563	78.315	3	-0.629	78.550	3	-0.492	74.360
UPC-POE	4	-0.982	72.928	3	-0.497	82.950	3	-0.678	75.032	3	-0.482	77.829	3	-0.383	74.095

(b) **Unseen entities**

	Data Coverage			Relevance			Correctness			Text Structure			Fluency		
Team Name	**Rank**	**Avg. Z**	**Avg. Raw**	**Rank**	**Avg. Z**	**Avg. Raw**	**Rank**	**Avg. Z**	**Avg. Raw**	**Rank**	**Avg. Z**	**Avg. Raw**	**Rank**	**Avg. Z**	**Avg. Raw**
AmazonAI (Shanghai)	1	0.170	94.098	1	0.215	95.713	1	0.201	92.933	1	0.295	93.498	1	0.284	90.550
WebNLG-2020-REF	1	0.230	95.178	2	0.066	93.389	1	0.263	94.207	1	0.277	92.190	1	0.310	90.508
OSU Neural NLG	1	0.303	96.033	1	0.173	94.941	1	0.237	93.489	1	0.319	92.941	1	0.340	90.423
FBConvAI *	2	0.140	92.780	2	0.098	93.644	1	0.173	91.669	1	0.308	92.605	1	0.293	90.006
NUIG-DSI	2	0.130	92.697	2	0.142	93.937	1	0.175	91.613	1	0.230	91.494	1	0.237	88.787
bt5	2	0.140	93.492	1	0.177	95.197	1	0.200	92.948	1	0.207	92.019	2	0.137	87.556
cuni-ufal	2	0.069	91.992	2	0.119	94.172	2	0.096	90.374	2	0.129	89.272	1	0.163	86.979
TGen	3	0.002	89.887	2	0.125	92.443	2	0.071	88.379	1	0.175	88.973	1	0.177	85.676
CycleGT	2	0.092	92.372	2	0.102	93.368	2	0.035	89.452	2	0.069	88.356	2	0.048	83.914
Baseline-FORGE2017	2	0.137	93.130	2	0.145	93.948	2	0.105	91.213	2	-0.032	87.542	2	-0.057	83.473
Baseline-FORGE2020	2	0.106	91.201	2	0.120	92.312	1	0.163	90.320	2	0.039	87.264	2	-0.007	82.414
DANGNT-SGU	1	0.284	95.897	1	0.172	94.872	1	0.210	93.142	3	-0.229	83.410	3	-0.182	77.992
RALI-Université de Montréal	1	0.253	95.805	1	0.176	95.678	1	0.119	92.054	3	-0.441	79.343	3	-0.387	74.554
ORANGE-NLG	5	-0.935	72.887	4	-1.225	71.728	4	-1.143	66.280	4	-0.617	75.743	4	-0.637	70.163
NILC	5	-0.970	72.234	4	-1.060	73.609	4	-1.098	65.856	4	-0.685	74.598	4	-0.721	67.330
Huawei Noah's Ark Lab	4	-0.586	80.004	3	-0.721	80.822	3	-0.743	73.427	4	-0.718	73.808	4	-0.700	67.308
UPC-POE	5	-0.932	73.157	3	-0.863	76.423	4	-1.075	67.586	4	-0.786	73.324	4	-0.828	66.358

(c) **Unseen categories**

Table 13: Human Evaluation results for **English** for **each test data type**. The systems are sorted by averaged Fluency raw scores. The colour intensity signifies final ranking in terms of averaged raw scores: more intense colour reflects higher performance for the specific criterion. * indicates late submission.

	DATA COVERAGE			RELEVANCE			CORRECTNESS			TEXT STRUCTURE			FLUENCY		
TEAM NAME	**Rank**	**Avg. Z**	**Avg. Raw**	**Rank**	**Avg. Z**	**Avg. Raw**	**Rank**	**Avg. Z**	**Avg. Raw**	**Rank**	**Avg. Z**	**Avg. Raw**	**Rank**	**Avg. Z**	**Avg. Raw**
bt5	1	0.312	95.630	1	0.174	95.385	1	0.340	95.594	1	0.219	95.745	1	0.232	93.088
cuni-ufal	1	0.203	93.155	1	0.077	93.306	2	0.101	90.382	1	0.218	96.073	1	0.213	92.921
FBConvAI *	1	0.133	92.339	2	0.027	93.491	2	0.080	90.779	2	0.079	93.764	2	0.063	90.248
WebNLG-2020-REF	1	0.230	94.000	2	0.065	93.636	2	0.109	90.630	2	-0.005	92.082	2	0.022	89.021
OSU Neural NLG	2	-0.422	82.836	2	-0.182	90.433	3	-0.181	84.830	2	0.019	92.958	2	-0.050	88.558
med	3	-0.467	82.230	2	-0.022	92.224	2	0.021	88.585	2	-0.077	91.309	2	-0.060	88.252
Huawei Noah's Ark Lab	2	-0.189	86.448	2	-0.060	91.761	2	-0.084	87.033	3	-0.183	89.515	3	-0.174	85.679
Baseline-FORGE2020	1	0.200	93.191	2	-0.079	91.294	4	-0.387	80.830	3	-0.270	87.645	3	-0.247	84.691

Table 14: Human Evaluation results for **Russian** RDF-to-text system submissions. The systems are sorted by averaged Fluency raw scores. The colour intensity signifies final ranking in terms of averaged raw scores: more intense colour reflects higher performance for the specific criterion. * indicates late submission.

Measure	1	2	3	4	5	6	7	8	9	10	11
1. BLEU NLTK	1.00										
2. METEOR	0.81	1.00									
3. chrF++	0.89	0.85	1.00								
4. TER	-0.87	-0.78	-0.85	1.00							
5. BERTScore F1	0.73	0.69	0.79	-0.74	1.00						
6. BLEURT	0.69	0.68	0.75	-0.77	0.76	1.00					
7. Correctness	0.35	0.29	0.41	-0.39	0.46	0.55	1.00				
8. Data Coverage	0.27	0.27	0.38	-0.31	0.39	0.49	0.71	1.00			
9. Fluency	0.38	0.31	0.4	-0.43	0.46	0.54	0.62	0.49	1.00		
10. Relevance	0.28	0.22	0.33	-0.32	0.38	0.47	0.72	0.7	0.51	1.00	
11. Text Structure	0.35	0.28	0.36	-0.39	0.44	0.51	0.56	0.45	0.8	0.51	1.00

(a) **English**

Measure	1	2	3	4	5	6	7	8	9	10
1. BLEU NLTK	1.00									
2. METEOR	0.91	1.00								
3. chrF++	0.92	0.92	1.00							
4. TER	-0.91	-0.91	-0.9	1.00						
5. BERTScore F1	0.83	0.83	0.93	-0.88	1.00					
6. Correctness	0.23	0.23	0.31	-0.24	0.31	1.00				
7. Data Coverage	0.20	0.2	0.32	-0.22	0.29	0.50	1.00			
8. Fluency	0.17	0.17	0.20	-0.22	0.26	0.42	0.31	1.00		
9. Relevance	0.14	0.14	0.17	-0.15	0.17	0.56	0.50	0.28	1.00	
10. Text Structure	0.16	0.16	0.19	-0.18	0.21	0.43	0.27	0.74	0.24	1.00

(b) **Russian**

Table 15: Pearson correlations for RDF-to-text task for both languages. All of them were statistically significant with p-value < 0.01.

grammaticality (Text Structure, Fluency) they score lower than neural approaches.

Russian. Table 14 shows the results of the human evaluation for the RDF-to-text systems in Russian. Similar to the automatic evaluation, the top-performing systems in all ratings are `bt5`, `cuni-ufal` and `FBConvAI`. Also, the ratings for the first two systems were significantly better than the ones for the ground-truth references for Relevance, Text Structure and Fluency. `bt5` also ranked higher than the references for Correctness. As described in Section 2.2, Russian data might have issues with fluency and correctness, so we attribute the lower ratings for references to the quality of the data. Interestingly, `Huawei Noah's Ark Lab` performed much worse on the human evaluation across all criteria compared to their automatic evaluation metric scores.

7 Correlation between Automatic and Human Evaluation Metrics

Tables 15a and 15b describe the sentence-level Pearson correlations of the evaluation metrics for the RDF-to-text task in English and Russian, respectively. Novel learned metrics, such as BERTScore and BLEURT, seem to correlate more with the human evaluation ratings than traditional token- and character-overlapping metrics, such as BLEU, METEOR, chrF++ and TER. For the English evaluation, BLEURT, the newest metric, was the one that best correlated with the human evaluation ratings, especially with Correctness and Fluency. For Russian, BERTScore was the automatic metric that best correlated with the human ratings, except for Data Coverage, with which chrF++ correlated the most. This latter was one of the character- and n-gram metrics that correlates more with human ratings.

8 Conclusion

This report described the data, participating systems, results and findings of the 2020 Bilingual, Bi-Directional WebNLG+ shared task. The shared task of this year involved two tasks: RDF-to-text and Text-to-RDF. In the following sections, we describe the main findings for each task conducted on this version of the shared task.

8.1 RDF-to-text

Similar to the WebNLG challenge of 2017, the RDF-to-text task consisted of verbalising sets of RDF triples. Different from the last version of this shared task, the task in this year was introduced in two languages: English and Russian. In total, we received 14 submissions for English and 6 for Russian.

Neural vs. Rule-based approaches. Looking at the results for the automatic and human evaluation, we could notice some differences between rule-based and neural approaches for data-to-text generation. The former models seem to automatically generate text comparable in quality with human texts in terms of adequacy, i.e., the generated texts express exactly the communicative goals contained in the input tripleset. On the other hand, novel neural approaches produce text comparable to human texts in terms of fluency.

Fine-tuned Large Language Models. Following a popular trend in Natural Language Processing, many of the participating neural approaches use fine-tuned large pre-trained language models, such as BART and T5. These systems, such as `Amazon AI`, `FBConvAI`, `OSU Neural NLG` and `bt5` were among the top-ranked systems and were rated high in terms of fluency and structure of the generated texts. T5 and BART were the large language models most frequently used by the participating systems. When comparing the use of both models, they seem to perform similarly in terms of fluency. However, the systems based on BART suffered a drop in performance in terms of data coverage. In contrast, the ones based on the T5 performed similarly to the rule-based approaches.

Memorisation vs. Generalisation. We evaluated the RDF-to-text systems in distinct data settings in order to verbalise *(i)* trials from semantic categories seen during training, *(ii)* trials from seen categories but with entities that were unseen during training and *(iii)* trials from categories fully unseen during training. We hypothesise that the former setting is the easiest, since the generation models would just have to memorise the content seen during training. On the other hand, in the latter the models would have to generalize the content learnt during training to unseen data. In fact, results confirmed that converting RDF triples from semantic categories seen during training to texts is easier than generating from unseen entities and semantic categories.

Automatic vs. Human Evaluation Metrics. We evaluated the participating systems using several traditional automated (e.g. BLEU, METEOR) metrics as well as novel learning-based evaluation ones (e.g. BERTScore, BLEURT) for text generation. The inclusion of all these metrics allowed us to investigate which metrics show stronger correlations with human ratings. We have observed that novel embedding-based metrics, such as BERTScore and BLEURT, achieved higher correlations with human ratings than traditional token-overlapping ones, such as BLEU and METEOR.

Parity with Human-Written References. Several systems achieved high performance in human evaluation across all the measured criteria and ended up in the same cluster with human references. Could we say that automatic systems generated almost human-like texts and "solved" the data-to-text WebNLG task? Those conclusions should be made with caution since the WebNLG dataset has its limitations. First, its vocabulary is relatively restricted; second, the dataset has a template-based structure where properties are lexicalised in a similar manner across texts. Given those drawbacks, the next edition of the shared task should aim for more complex and naturally occurring texts verbalising RDF triples. Specifically, for Russian, we will need to collect better data to measure parity with automatic systems.

Overall, modelling language is always a moving target. So we should strive for better and versatile datasets and evaluation settings.

8.2 Text-to-RDF

The Text-to-RDF task was a new challenge for WebNLG. Natural language texts had to be converted to RDF triples from the Semantic Web. Similar to RDF-to-text, this task was introduced in two languages: English and Russian. In total, **three**

teams participated in the task. One team participated in both the English and Russian version of the task, whereas two participated only in the English version.

The evaluation was only performed using automatic metrics (F1, Precision, Recall) on four different levels (Exact, Ent_Type, Partial, Strict). The results on these metrics for English submissions show that all of them were able to outperform the baseline based on Stanford CoreNLP's Open Information Extraction module (Manning et al., 2014). In particular, `Amazon AI (Shanghai)` and `bt5` performed well compared to this baseline. At the same time, when comparing the results per data type, a drop in performance was observed for all systems when tested for the sets with unseen categories and unseen entities, indicating that all submitted semantic parsing systems struggle with generalisation to unseen entities and categories. However, note that the scores for `Amazon AI (Shanghai)` and `bt5` still stayed relatively high when being tested on the unseen categories and unseen entities.

Only `bt5` participated in the Russian version of the task and achieved very high scores compared to the baseline. However, we emphasise that the Russian test set included only seen entities and categories. Hence, it is not clear how well the Russian `bt5` system would be able to generalize to unseen entities and categories.

Acknowledgments

We are grateful to Elena Khasanova for developing and monitoring the crowdsourcing pipeline for Russian data collection during her internship at LORIA in summer 2019, and to Stamatia Dasiopoulou for her work on the baseline's interface between RDF and predicate-argument structures. We gratefully acknowledge the support of the French National Research Agency (Gardent; award ANR-20-CHIA-0003, XNLG "Multi-lingual, Multi-Source Text Generation") and of the European Commission in the context of its H2020 Program under the grant numbers 870930-RIA, 779962-RIA, 825079-RIA, 786731-RIA at Universitat Pompeu Fabra. Research funded by the German Federal Ministry of Economics and Technology (BMWI) in the project RAKI (no. 01MD19012D). This work also has been supported by the German Federal Ministry for Economic Affairs and Energy (BMWi) within the project SPEAKER under the grant no 01MK20011U; and by the Coordination for the Improvement of Higher Education Personnel in Brazil (CAPES) under the grant 88887.508597/2020-00.

References

Oshin Agarwal, Mihir Kale, Heming Ge, Siamak Shakeri, and Rami Al-Rfou. 2020. Machine translation aided bilingual data-to-text generation and semantic parsing. In *Proceedings of the 3rd WebNLG Workshop on Natural Language Generation from the Semantic Web (WebNLG+ 2020)*, Dublin, Ireland (Virtual). Association for Computational Linguistics.

Olga Babko-Malaya. 2005. *Propbank Annotation Guidelines*.

David Batista. 2018. Named-entity evaluation metrics based on entity-level. Retrieved from `http://www.davidsbatista.net/blog/2018/05/09/Named_Entity_Evaluation/` on November 14, 2020.

Pavel Blinov. 2020. Semantic triples verbalization with generative pre-training model. In *Proceedings of the 3rd WebNLG Workshop on Natural Language Generation from the Semantic Web (WebNLG+ 2020)*, Dublin, Ireland (Virtual). Association for Computational Linguistics.

Boxing Chen and Colin Cherry. 2014. A systematic comparison of smoothing techniques for sentence-level BLEU. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pages 362–367, Baltimore, Maryland, USA. Association for Computational Linguistics.

Oriol Domingo Roig, David Bergés Lladó, Roser Cantenys Sabà, Roger Creus Castanyer, and José Adrián Rodríguez Fonollosa. 2020. The upc rdf-to-text system at webnlg challenge 2020. In *Proceedings of the 3rd WebNLG Workshop on Natural Language Generation from the Semantic Web (WebNLG+ 2020)*, Dublin, Ireland (Virtual). Association for Computational Linguistics.

Claire Gardent, Anastasia Shimorina, Shashi Narayan, and Laura Perez-Beltrachini. 2017a. Creating training corpora for NLG micro-planners. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 179–188, Vancouver, Canada. Association for Computational Linguistics.

Claire Gardent, Anastasia Shimorina, Shashi Narayan, and Laura Perez-Beltrachini. 2017b. The WebNLG challenge: Generating text from RDF data. In *Proceedings of the 10th International Conference on Natural Language Generation*, pages 124–133, Santiago de Compostela, Spain. Association for Computational Linguistics.

Qipeng Guo, Zhijing Jin, Ning Dai, Xipeng Qiu, Xiangyang Xue, David Wipf, and Zheng Zhang. 2020a.

Improving data-to-text generation by pre-training and planning. In *Proceedings of the 3rd WebNLG Workshop on Natural Language Generation from the Semantic Web (WebNLG+ 2020)*, Dublin, Ireland (Virtual). Association for Computational Linguistics.

Qipeng Guo, Zhijing Jin, Xipeng Qiu, Weinan Zhang, David Wipf, and Zheng Zhang. 2020b. Cyclegt: Unsupervised graph-to-text and text-to-graph generation via cycle training. In *Proceedings of the 3rd WebNLG Workshop on Natural Language Generation from the Semantic Web (WebNLG+ 2020)*, Dublin, Ireland (Virtual). Association for Computational Linguistics.

Zdenek Kasner and Ondrej Dusek. 2020. Train hard, finetune easy: Multilingual denoising for rdf-to-text generation. In *Proceedings of the 3rd WebNLG Workshop on Natural Language Generation from the Semantic Web (WebNLG+ 2020)*, Dublin, Ireland (Virtual). Association for Computational Linguistics.

Natthawut Kertkeidkachorn and Hiroya Takamura. 2020. Text-to-text pre-training model with plan selection for rdf-to-text generation. In *Proceedings of the 3rd WebNLG Workshop on Natural Language Generation from the Semantic Web (WebNLG+ 2020)*, Dublin, Ireland (Virtual). Association for Computational Linguistics.

Guy Lapalme. 2020. Rdfg: a symbolic approach for generating text from rdf triples. In *Proceedings of the 3rd WebNLG Workshop on Natural Language Generation from the Semantic Web (WebNLG+ 2020)*, Dublin, Ireland (Virtual). Association for Computational Linguistics.

Alon Lavie and Abhaya Agarwal. 2007. METEOR: An automatic metric for MT evaluation with high levels of correlation with human judgments. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 228–231, Prague, Czech Republic. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation.

Edward Loper and Steven Bird. 2002. NLTK: the natural language toolkit. *CoRR*, cs.CL/0205028.

Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60, Baltimore, Maryland. Association for Computational Linguistics.

Simon Mille, Stamatia Dasiopoulou, Beatriz Fisas, and Leo Wanner. 2019a. Teaching FORGe to verbalize DBpedia properties in Spanish. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 473–483, Tokyo, Japan. Association for Computational Linguistics.

Simon Mille, Stamatia Dasiopoulou, and Leo Wanner. 2019b. A portable grammar-based nlg system for verbalization of structured data. In *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing*, pages 1054–1056. ACM.

Sebastien Montella, Betty Fabre, Lina Maria Rojas-Barahona, Johannes Heinecke, and Tanguy Urvoy. 2020. Denoising pre-training and data augmentation strategies for enhanced rdf verbalization with transformers. In *Proceedings of the 3rd WebNLG Workshop on Natural Language Generation from the Semantic Web (WebNLG+ 2020)*, Dublin, Ireland (Virtual). Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Nivranshu Pasricha, Mihael Arcan, and Paul Buitelaar. 2020. Nuig-dsi at the webnlg+ challenge: Leveraging transfer learning for rdf-to-text generation. In *Proceedings of the 3rd WebNLG Workshop on Natural Language Generation from the Semantic Web (WebNLG+ 2020)*, Dublin, Ireland (Virtual). Association for Computational Linguistics.

Maja Popović. 2017. chrF++: words helping character n-grams. In *Proceedings of the Second Conference on Machine Translation*, pages 612–618, Copenhagen, Denmark. Association for Computational Linguistics.

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. 2020. Stanza: A python natural language processing toolkit for many human languages.

A. Radford, Jeffrey Wu, R. Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text

transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Ehud Reiter and Robert Dale. 2000. *Building natural language generation systems*. Cambridge university press.

Isabel Segura-Bedmar, Paloma Martínez, and María Herrero-Zazo. 2013. SemEval-2013 task 9 : Extraction of drug-drug interactions from biomedical texts (DDIExtraction 2013). In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 341–350, Atlanta, Georgia, USA. Association for Computational Linguistics.

Thibault Sellam, Dipanjan Das, and Ankur Parikh. 2020. BLEURT: Learning robust metrics for text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7881–7892, Online. Association for Computational Linguistics.

Rico Sennrich, Alexandra Birch, Anna Currey, Ulrich Germann, Barry Haddow, Kenneth Heafield, Antonio Valerio Miceli Barone, and Philip Williams. 2017. The University of Edinburgh's Neural MT Systems for WMT17. In *Proceedings of the Second Conference on Machine Translation, Volume 2: Shared Task Papers*, Copenhagen, Denmark.

Anastasia Shimorina, Elena Khasanova, and Claire Gardent. 2019. Creating a corpus for Russian data-to-text generation using neural machine translation and post-editing. In *Proceedings of the 7th Workshop on Balto-Slavic Natural Language Processing*, pages 44–49, Florence, Italy. Association for Computational Linguistics.

Matthew Snover, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *In Proceedings of Association for Machine Translation in the Americas*, pages 223–231.

Marco Antonio Sobrevilla Cabezudo and Thiago Alexandre Salgueiro Pardo. 2020. Nilc at webnlg+: Pretrained sequence-to-sequence models on rdf-to-text generation. In *Proceedings of the 3rd WebNLG Workshop on Natural Language Generation from the Semantic Web (WebNLG+ 2020)*, Dublin, Ireland (Virtual). Association for Computational Linguistics.

Trung Tran and Dang Tuan Nguyen. 2020. Webnlg 2020 challenge: Semantic template mining for generating references from rdf. In *Proceedings of the 3rd WebNLG Workshop on Natural Language Generation from the Semantic Web (WebNLG+ 2020)*, Dublin, Ireland (Virtual). Association for Computational Linguistics.

Xintong, Aleksandre Maskharashvili, Symon Jory Stevens-Guille, and Michael White. 2020. Leveraging large pretrained models for webnlg 2020. In *Proceedings of the 3rd WebNLG Workshop on Natural Language Generation from the Semantic Web (WebNLG+ 2020)*, Dublin, Ireland (Virtual). Association for Computational Linguistics.

Zixiaofan Yang, Arash Einolghozati, Hakan Inan, Angela Fan, Pinar Donmez, and Sonal Gupta. 2020. Improving text-to-text pre-trained models for the graph-to-text task. In *Proceedings of the 3rd WebNLG Workshop on Natural Language Generation from the Semantic Web (WebNLG+ 2020)*, Dublin, Ireland (Virtual). Association for Computational Linguistics.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

Chao Zhao, Marilyn Walker, and Snigdha Chaturvedi. 2020. Bridging the structural gap between encoding and decoding for data-to-text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, volume 1.

Giulio Zhou and Gerasimos Lampouras. 2020. Webnlg challenge 2020: Language agnostic delexicalisation for multilingual rdf-to-text generation. In *Proceedings of the 3rd WebNLG Workshop on Natural Language Generation from the Semantic Web (WebNLG+ 2020)*, Dublin, Ireland (Virtual). Association for Computational Linguistics.

Appendix A Example task for human evaluation experiments on MTurk

Please (i) follow the instructions, (ii) be honest and fair in your judgements, (iii) try to be as correct as possible in your conclusions. For example, the text would generally get a score higher than 0 for Correctness if at least some objects in it are introduced correctly. Similarly, the text would not be rated with 100 for Correctness if at least one object is not introduced correctly.

Task Instructions
You are given a piece of data and a text that describes data.
Below you will find statements that relate to the text.
Please rate each of these statements by moving the slider along the scale where 0 stands for 'I do not agree', 100 stands for 'I fully agree'.

To learn more about the task, its details and examples, click on 'View Detailed Instructions' below!

View Detailed Instructions

DATA

Subject	Predicate	Object
Agnes Kant	nationality	Netherlands
Netherlands	leader	Mark Rutte
Agnes Kant	office	"Member of the House of Representatives"
Agnes Kant	party	Socialist Party (Netherlands)

DESCRIPTION

Agnes Kant is a member of the Socialist Party of the Netherlands where Mark Rutte is the leader

How well do you agree with the following statements?

Data Coverage: The text contains all predicates from the data and does not miss any predicates shown in the data.

Relevance: The text contains only known/relevant predicates, which are found in the data. The text does not contain any unknown/irrelevant/unrecognizable predicates.

Correctness: When describing information about relevant predicates (those, which are in both data and text), the text depicts them with correct/proper objects. Also, the text correctly introduces the Subject.

Text Structure: The text is written in good English language, i.e. it is free from grammatical errors and well-structured.

Fluency: The text sounds logically correct and forms a coherent whole. There are no parts of the text you would change to make it sound better. The text forms a nice narrative.

Write you feedback in the field below if you have any (not necessary):

Your feedback here...

Submit

Appendix B Example task for human evaluation experiments on Yandex.Toloka

Пожалуйста, (1) следуйте инструкциям, (2) будьте честны и справедливы в своих суждениях касательно задания, (3) постарайтесь быть максимально корректны в своих оцениваниях текстов.

Инструкции
Вам представлены данные в виде таблицы и текст, который описывает эти данные. В данных есть субъекты, объекты и отношения.
Ниже данных Вам представлены суждения для оценивания текста.
Пожалуйста, оцените каждое суждение по шкале от 0 до 100, где 0 обозначает, что Вы не согласны с суждением, а 100 обозначает, что Вы полностью согласны с суждением. Используйте соответсвующий ползунок, чтобы определиться с выбором оценки по критерию.

Чтобы узнать больше про задание, прочитать детали и примеры, кликните на иконку "Инструкция" в правом верхнем углу страницы!

Данные

Субъект	Отношение	Объект
Олдрин, Базз	экспедиция	Аполлон-11

Текст:

> Олдрин Базз принял участие в экспедиции Аполлон-11

Насколько Вы согласны со следующими утверждениями?

Покрытие Данных: описывает ли текст все отношения, которые присутствуют в данных?

Релевантность: описывает ли текст только те отношения, которые присутствуют в данных, и никакие иначе?

Корректность (Правильность): описывает ли текст отношения с использованием правильных объектов, согласно данным? Правильно ли описан субъект в тексте?

Структура Текста: Текст написан на хорошем русском языке, в нем отсутствуют грамматические ошибки, а предложения построены правильно.

Плавность Текста: Текст логически выверен и является связным целым. В тексте отсутствуют части, которые требовали бы изменений, чтобы улучшить естественность текста.

Поделитесь своим отзывом и комментариями к заданию (если у Вас таковые имеются):